

**IN THE CLAIMS:**

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. - 68. (Cancelled)

69. (New) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

- displaying a node in a graphical program;
- receiving first user input invoking display of a plurality of functions for the node;
- displaying the plurality of functions for the node in response to the first user input;

- receiving second user input selecting a function from the plurality of functions;
- determining graphical program code based on the second user input, wherein the determined graphical program code comprises a graphical representation of an implementation of the selected function, and wherein the determined graphical program code is executable to provide functionality in accordance with the selected function;

- associating the determined graphical program code with the node, wherein, when the node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

70. (New) The memory medium of claim 69, wherein the node has a first node icon which is displayed in the graphical program, and wherein the first node icon has a first appearance, wherein the program instructions are further executable to perform:

- changing the first node icon to a second appearance based on the second user input, wherein said changing the first node icon to a second appearance includes displaying an image corresponding to the selected function.

71. (New) The memory medium of claim 70,

wherein said changing the first node icon to a second appearance comprises replacing the first node icon with a second node icon.

72. (New) The memory medium of claim 69, wherein, prior to said associating the determined graphical program code with the node, the node does not have any associated graphical program code.

73. (New) The memory medium of claim 69,

wherein, prior to said associating the determined graphical program code with the node, the node has associated default graphical program code in accordance with a default function for the node, and wherein the default graphical program code implements a first functionality; and

wherein said associating the determined graphical program code with the node comprises replacing the default graphical program code with the determined graphical program code.

74. (New) The memory medium of claim 69,

wherein said receiving first user input comprises receiving the first user input to the node; and

wherein said receiving second user input comprises receiving the second user input to the node.

75. (New) The memory medium of claim 69,

wherein said displaying the plurality of functions for the node in response to the first user input comprises:

displaying a plurality of function classes for the node; and

in response to user input selecting a function class, displaying the plurality of functions, wherein the plurality of functions are in the selected function class.

76. (New) The memory medium of claim 69,

wherein the node is a data acquisition (DAQ) node;

wherein the plurality of functions for the node comprise a plurality of DAQ functions;

wherein, prior to said associating, the DAQ node comprises one of:

- a generic read node;
- a generic write node;
- a generic channel creation node;
- a generic timing node; or
- a generic triggering node; and

wherein, after said associating, the DAQ node comprises one of:

- a specific read node in accordance with the selected function;
- a specific write node in accordance with the selected function;
- a specific channel creation node in accordance with the selected function;
- a specific timing node in accordance with the selected function; or
- a specific triggering node in accordance with the selected function.

77. (New) A computer-implemented method for configuring a graphical program node, comprising:

- displaying a node in a graphical program;
- receiving first user input invoking display of a plurality of functions for the node;
- displaying the plurality of functions for the node in response to the first user input;

- receiving second user input selecting a function from the plurality of functions;

- determining graphical program code based on the second user input, wherein the determined graphical program code comprises a graphical representation of an implementation of the selected function, and wherein the determined graphical program code is executable to provide functionality in accordance with the selected function;

- associating the determined graphical program code with the node, wherein, when the node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

78. (New) The method of claim 77, wherein the node has a first node icon which is displayed in the graphical program, the method further comprising:

changing the first node icon to a second appearance based on the second user input, wherein said changing the first node icon to a second appearance includes displaying an image corresponding to the selected function.

79. (New) The method of claim 78,

wherein said changing the first node icon to a second appearance comprises replacing the first node icon with a second node icon.

80. (New) The method of claim 77, wherein, prior to said associating the determined graphical program code with the node, the node does not have any associated graphical program code.

81. (New) The method of claim 77,

wherein, prior to said associating the determined graphical program code with the node, the node has associated default graphical program code in accordance with a default function for the node, and wherein the default graphical program code implements a first functionality; and

wherein said associating the determined graphical program code with the node comprises replacing the default graphical program code with the determined graphical program code.

82. (New) The method of claim 77,

wherein said receiving first user input comprises receiving the first user input to the node; and

wherein said receiving second user input comprises receiving the second user input to the node.

83. (New) The method of claim 77,

wherein said displaying the plurality of functions for the node in response to the first user input comprises:

displaying a plurality of function classes for the node; and

in response to user input selecting a function class, displaying the plurality of functions, wherein the plurality of functions are in the selected function class.

84. (New) The method of claim 77,

wherein the node is a data acquisition (DAQ) node;

wherein the plurality of functions for the node comprise a plurality of DAQ functions;

wherein, prior to said associating, the DAQ node comprises one of:

a generic read node;

a generic write node;

a generic channel creation node;

a generic timing node; or

a generic triggering node; and

wherein, after said associating, the DAQ node comprises one of:

a specific read node in accordance with the selected function;

a specific write node in accordance with the selected function;

a specific channel creation node in accordance with the selected function;

a specific timing node in accordance with the selected function; or

a specific triggering node in accordance with the selected function.

85. (New) A computer-accessible memory medium that stores program instructions executable by a processor to perform:

displaying a node in a graphical program;

receiving first user input invoking display of a plurality of functions for the node;

displaying the plurality of functions for the node in response to the first user input;

receiving second user input selecting a function from the plurality of functions;  
determining a second node based on the selected function, wherein the second node comprises a graphical representation of an implementation of the selected function, and wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;  
replacing the node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the graphical program code of the second node executes to provide the functionality in accordance with the selected function.

86. (New) The memory medium of claim 85, wherein the node comprises a first node icon, and wherein said displaying the node comprises displaying the first node icon, and wherein the second node comprises:

- the first node icon and the graphical program code; or
- a second node icon and the graphical program code.

87. (New) The memory medium of claim 85, wherein the node and/or the second node is one or more of:

- polymorphic;
- function switchable; or
- function class switchable.

88. (New) The memory medium of claim 85,  
wherein the node is a data acquisition (DAQ) node;  
wherein the DAQ node comprises one of:

- a generic read node;
- a generic write node;
- a generic channel creation node;
- a generic timing node; or
- a generic triggering node; and

wherein the second node comprises a corresponding one of:

- a specific read node in accordance with the selected function;
- a specific write node in accordance with the selected function;
- a specific channel creation node in accordance with the selected function;
- a specific timing node in accordance with the selected function; or
- a specific triggering node in accordance with the selected function.

89. (New) A computer-implemented method for configuring a graphical program node, comprising:

- displaying a node in a graphical program;
- receiving first user input invoking display of a plurality of functions for the node;
- displaying the plurality of functions for the node in response to the first user input;
- receiving second user input selecting a function from the plurality of functions;
- determining a second node based on the selected function, wherein the second node comprises a graphical representation of an implementation of the selected function, and wherein the second node comprises graphical program code executable to provide functionality in accordance with the selected function;
- replacing the node in the graphical program with the second node, wherein, when the second node in the graphical program executes, the determined graphical program code executes to provide the functionality in accordance with the selected function.

90. (New) The method of claim 89, wherein the node comprises a first node icon, and wherein said displaying the node comprises displaying the first node icon, and wherein the second node comprises:

- the first node icon and the graphical program code; or
- a second node icon and the graphical program code.

91. (New) The memory medium of claim 89, wherein the node and/or the second node is one or more of:

- polymorphic;

function switchable; or  
function class switchable.

92. (New) The memory medium of claim 89,  
wherein the node is a data acquisition (DAQ) node;  
wherein the DAQ node comprises one of:  
    a generic read node;  
    a generic write node;  
    a generic channel creation node;  
    a generic timing node; or  
    a generic triggering node; and  
wherein the second node comprises a corresponding one of:  
    a specific read node in accordance with the selected function;  
    a specific write node in accordance with the selected function;  
    a specific channel creation node in accordance with the selected function;  
    a specific timing node in accordance with the selected function; or  
    a specific triggering node in accordance with the selected function.